

ALGORITHME DE CONTROLE REACTIF DE LA CONGESTION SUR UN NŒUD ATM

DADA Abdelmalek ⁽¹⁾ et BENNIA Abdelhak ⁽²⁾

⁽¹⁾ Direction des transmissions nationales, Constantine, Algérie.

abdelmalek.dada@wanadoo.dz

⁽²⁾ Département d'Electronique, Université Mentouri de Constantine.

1- Introduction

L'ATM est un ensemble de compromis techniques, parmi ceux ci, le choix essentiel a été le découpage de toutes les informations en cellules de 53 octets chacune (5 Octets pour l'en-tête et 48 pour la charge).

Charge : 48 octets	En-tête : 5
--------------------	-------------

Figure 1 : La cellule ATM.

La commutation par cellules de taille fixe, est la base de l'architecture ATM et constitue l'innovation de cette technique. Chaque cellule est une entité dont l'en-tête assure l'acheminement. De plus, toutes les cellules d'une même communication empruntent la même route à travers le réseau ATM : cette contrainte assure la conservation de l'ordre d'arrivée des cellules conformément à celui de l'émission et facilite la détection de perte des cellules [1,2,3].

2- ABR : Available Bit Rate

L'ATM propose plusieurs catégories de service capables de répondre aux différents besoins de la majorité des applications multimédias : CBR : Constant Bit Rate, VBR : Variable Bit Rate, UBR : Unspecified Bit Rate, GFR : Guaranteed Frame Rate, ABR :

Available Bit Rate [1,2,3].

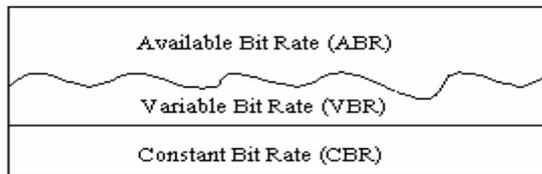


Figure 2 : Répartition de la bande passante ATM.

L'ABR ; une des classes de service proposées par la technologie ATM, destinée à des applications qui exploitent la bande passante de la liaison demeurant disponible par des services plus exigeants (Figure 2), son étude soulève de nombreuses difficultés techniques, parmi celles-ci ; l'apparition de la congestion sur un des nœuds de la liaison [3,4].

3- Problématique

La conception et le dimensionnement des liaisons et des circuits de commutation ATM et leurs architectures, visent en premier lieu à réduire au maximum les retards et l'effet de la congestion. Le problème est localisé au niveau des cellules ayant déjà été acceptées sur le multiplex et qui exigent une opération de commutation sur une artère de sortie pour arriver à destination. Trop attendre un traitement par manque d'espace mémoire sur les buffers, conduit à la destruction systématique des cellules non admises, ceci constitue un handicap pour la technologie ATM [1,2,4,6].

Pour une connexion donnée, les cellules peuvent traverser de nombreux nœuds intermédiaires du réseau avant d'atteindre leur destination, mais c'est en général à un nœud précis que peut apparaître le problème de la congestion : goulot d'étranglement de la connexion, là où la bande passante est la plus faible. Les procédures de contrôle doivent estimer les ressources disponibles au niveau de ce nœud et adapter le trafic soumis au réseau en fonction des paramètres mesurés.

4-Les contrôles de la congestion

Deux types de contrôles peuvent être envisagés : un contrôle préventif et un contrôle réactif. Comme son nom le suggère, le contrôle préventif consiste à prendre des mesures

a priori, visant à minimiser les chances de l'apparition d'une congestion. Par exemple, les réservations de la bande passante ou des mémoires dans le réseau font partie du contrôle préventif [1,2].

Le fait que la connexion déclare le débit d'émission est aussi utilisé pour des actions préventives. La philosophie du contrôle préventif est de protéger le réseau contre les rafales de données, et ce afin de pouvoir assurer une qualité de service satisfaisante. À l'inverse du contrôle préventif, le contrôle réactif agit en fonction de l'évolution de l'état du réseau. Dans ce cas, le réseau accepte toutes les connexions et tant qu'il n'y a pas de congestion, celles ci peuvent émettre le débit qui leur convient. La mesure de la congestion peut se faire de plusieurs manières (en mesurant les pertes, les délais, le remplissage des buffers, etc.) par le réseau ou par les équipements terminaux. Si une congestion se déclare la source diminue son débit. Le principe du contrôle réactif est de pouvoir partager les ressources par le maximum de connexions pour optimiser l'utilisation du réseau. De ce fait, ce type de contrôle peut être approprié aux connexions de données qui peuvent s'adapter aux conditions du réseau. Cependant, il n'est pas adapté aux connexions qui nécessitent une garantie de qualité de service tels que les services vidéo en temps réel [3,4].

Le contrôle de trafic dans le réseau ATM se base sur la déclaration de paramètres de trafic et la conformité du trafic à ces paramètres. En effet, lors de la demande d'établissement d'une connexion, l'utilisateur déclare un certain nombre de paramètres décrivant son trafic [1,2,5,8]. Ces paramètres sont utilisés par le réseau pour prendre des mesures préventives afin d'éviter les congestions. Outre les paramètres de trafic, l'utilisateur spécifie les valeurs désirées des attributs de qualité de service (QoS) requise par la connexion. Par exemple, le délai et le taux de perte sont des attributs de la qualité de service [5,8]. Le réseau décide alors d'accepter ou de refuser la connexion selon qu'il estime pouvoir, ou non, satisfaire les contraintes de QoS spécifiées. La procédure qui lui permet de décider s'appelle procédure de contrôle d'admission ou CAC (Connection Admission Control). Le contrôle est fait à la volée et en temps réel, et les cellules non conformes peuvent être rejetées ou admises, selon la politique de l'opérateur. Le réseau définit une politique d'allocation et de partage de ses ressources entre les connexions dont le but est de respecter la qualité de service négociée tout en maximisant l'utilisation des ressources du réseau.

Le travail présenté à travers cet article s'inscrit dans le cadre du contrôle réactif et fait appel à une approche très intelligente qui consiste à faire circuler sur le réseau une

et de l'ITU [3,5,6,7,8] en vue d'une exploitation idéale de cette source périodique d'informations sur l'état général du réseau.

requête périodique identifiée par la source et la destination, ce principe est appelé ressource management RM et fait l'objet de recommandations de la part du ATM-Forum

5- Aperçu sur les travaux antérieurs

Plusieurs méthodes, faisant appel à divers mécanismes techniques; ont été proposées pour approcher le phénomène de la congestion et présentées dans de nombreux travaux de recherches [3,5,6,7,8,10,13], allant de la simple implémentation d'algorithmes à la révision complète de l'architecture du réseau, commutateurs et mémoires tampons. Le choix, comme expliqué dans [3,5], dépend principalement de la durée de présence de la congestion sur le réseau.

Les références [3,5,13] ont passé en revue ces méthodes avec des simulations comparatives mettant en évidence les avantages et surtout les inconvénients de chaque méthode.

Parmi les méthodes de contrôle de la congestion qui visent à renforcer les mesures préventives prises avant l'admission des cellules sur le réseau ; on peut citer le rejet sélectif, la limitation du débit par fenêtrage, l'échange périodique d'informations entre source et réseau.

Le présent article a retenu l'approche algorithmique, vu sa capacité à répondre aux cas extrêmes de congestion (longue durée, limitation de la bande passante, retards, etc.) et également pour la souplesse et la simplicité d'implémentation qu'elle offre.

6- Algorithme

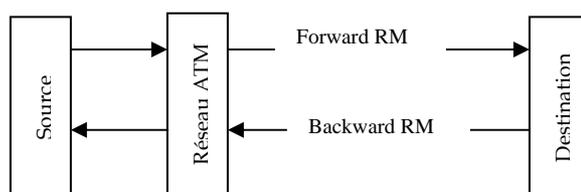


Figure 3 : Principe du RM.

L'approche Resource Management RM [3,5,11,12], est un modèle d'algorithme basé sur le principe de l'envoi d'un paquet FRM (Forward RM) appelé requête et la réception d'un paquet BRM (Backward RM). Le procédé est totalement indépendant du type d'informations à transporter à travers le réseau, seul le trafic émis au niveau de la source est soumis au contrôle du RM.

Le procédé repose sur un échange périodique d'information entre la source et la destination. La source émet un paquet de requêtes FRM toutes les T unités, la période T étant connue aussi par la destination. Emettre une requête consiste simplement à marquer un paquet de données de manière spéciale, pour qu'il soit reconnu par la destination comme étant une requête. Dès que la destination reçoit une requête, elle envoie en retour un paquet d'information BRM sur le dernier intervalle de mesure. A la réception de ce paquet de réponse, la source adapte en conséquence le trafic soumis au réseau, ce mécanisme est illustré par la figure 3.

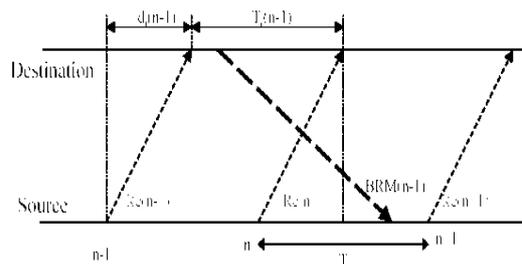


Figure 4 : Chronologie de l'algorithme RM.

Le contrôle au niveau de la source porte sur le débit $R_e(n)$, variable autorisée par l'algorithme dans l'intervalle n , entre l'émission de la n ème et de la $(n+1)$ -ième requête (Figure 4). Le n ème paquet de BRM, en réponse à la n ème requête, contient les informations sur le $(n-1)$ -ième intervalle de mesure, mais ne peut être pris en compte par la source que pour l'intervalle $(n+1)$. Nous supposons que la source est saturée (pratiquement, il y a toujours des données à transmettre), et $R_e(n)$ est dans ce cas simplement le taux émis par la source durant tout l'intervalle n . Le taux d'émission $R_e(n+1)$ est donc défini en fonction des mesures de l'intervalle $(n-1)$.

7- Phases du contrôle

Le contrôle du taux $R_e(n)$ se fait suivant un algorithme exécuté au niveau de la source. Notons $d_t(n)$ le délai total du nième paquet de requête de la source jusqu'à la destination. Nous supposons que $d_t(n)$ est la somme d'un délai de propagation fixe $d_p(n)$ et d'un délai d'attente $d_q(n)$ dans les buffers du réseau qui varie selon l'encombrement : $d_t(n)=d_p+d_q(n)$. Soit $T_r(n-1)$ l'intervalle de temps au niveau de la destination qui sépare la réception du $(n-1)$ -ième et du n -ième paquet de requête. L'intervalle de temps qui sépare l'émission de ces deux paquets par la source est égal à T , on a donc

$$d_t(n-1) + T_r(n-1) = T + d_t(n) \quad (1)$$

ce qui entraîne, en notant $\Delta T(n-1)=T_r(n-1)-T$ la différence des intervalles de temps entre source et destination :

$$d_q(n)=d_q(n-1)+ \Delta T(n-1) \quad (2)$$

L'équation précédente donne le moyen de calculer une valeur estimée $d_e(n)$ du délai $d_q(n)$ en mesurant les intervalles de temps $T_r(n-1)$. Avec une valeur initiale $d_e(0)=0$, on pose :

$$d_e(n)=d_e(n-1)+ \Delta T(n-1) \quad (3)$$

la valeur estimée $d_e(n)$ est alors égale à $d_q(n)$ à une constante d'erreur près, car les deux suivent les mêmes variations.

L'algorithme de la source est exécuté au début de chaque intervalle de mesure, au moment d'envoyer un nouveau paquet de requête. On définit deux modes de fonctionnement $M1$ et $M2$, selon que le réseau semble être chargé ou non. Le mode $M1$ est le mode normal d'utilisation, lorsque le réseau est en bonne condition et que le délai estimé $d_e(n)$ ne dépasse pas un certain seuil de tolérance T_q , avec $0 < T_q < T$. si la valeur $d_e(n)$ est stable ou diminue, la source est alors autorisée à augmenter le taux d'émission $R_e(n)$ d'une valeur R_{inc} . Lorsque $d_e(n)$ dépasse le seuil T_q , le mode devient $M2$ et le taux d'émission autorisé $R_e(n)$ est réduit, la nouvelle valeur $R_e(n+1)$ étant inférieure à $R_e(n)$ de la différence $R_e(n)$ et $R_e(n-1)$, avec un paramètre $\theta < 1$.

Si le paquet de BRM n'est pas encore reçu au moment d'effectuer ces opérations, et donc si le taux $R_r(n-1)$ n'est pas connu, des mesures plus sévères sont prises pour réduire le taux $R_e(n)$. Nous supposons dans la suite que cette situation ne se produit pas. L'algorithme exécuté au moment de l'émission du paquet de requête $n+1$ est le suivant :

- Si $d_e(n) \leq d_e(n-1)$ et le mode est $M1$, alors $R_e(n+1)=R_e(n)+R_{inc}$, et le mode reste $M1$. Dans ce cas, on a $d_e(n-1) < T_q$ car le mode de l'intervalle précédent est $M1$, et donc $d_e(n) < T_q$.

- Si $d_e(n) > Tq$, ou si le n -ième paquet de BRM n'a pas été reçu, alors, $\text{mode} = M2$ et

$$R_e(n+1) = R_e(n) - \theta \min(R_e(n), R_e(n-1)).$$
- Dans tous les autres cas, $R_e(n+1) = R_e(n)$ et le $\text{mode} = M1$.

8- Contribution

Sous le contrôle de l'algorithme Resource Management, on considère un modèle simple de connexion, illustré par la figure 2. Nous représentons un seul nœud intermédiaire, qui correspond au point de congestion sur cette connexion.

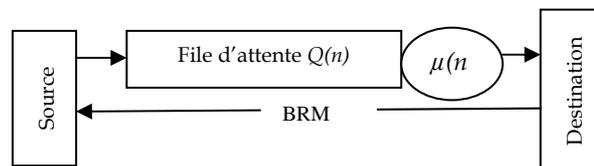


Figure 5 : Nœud intermédiaire de simulation.

Soit $\mu(n)$ le taux de service disponible au nœud intermédiaire pendant le n ème intervalle de mesure. Nous supposons que $\mu(n)$ est constant durant tout cet

intervalle, entre l'arrivée dans la file d'attente des paquets de requête n et $n+1$, mais peut varier d'un intervalle au suivant : la bande passante disponible pour la connexion considérée varie en fonction des autres connexions qui passent par ce nœud. Nous représentons les variations de $\mu(n)$ par une marche aléatoire dans un intervalle $[\mu_m, \mu_M]$. Nous faisons l'hypothèse que la valeur estimée $d_e(n)$ du délai d'attente $d_q(n)$ du n ème paquet de requête est exacte, et nous posons $d(n) = d_e(n) = d_q(n)$. Soit $Q(n)$ la longueur de la file d'attente au moment où le n ème paquet de requête y est admis, on a donc :

$$d(n) = Q(n) / \mu(n) \quad (4)$$

Il va de soit, que les valeurs de $R_e(n)$, $\mu(n)$ et $Q(n)$ sont à valeurs dans N (nombre entiers), on peut compter par exemple $R_e(n)$ et $\mu(n)$ en nombres entiers de cellules/s et $Q(n)$ en nombre de cellules. L'évolution du système décrite ci-dessus, avec le calcul de $R_e(n+1)$, $\mu(n+1)$ et $Q(n+1)$ en fonction du passé du système jusqu'au temps n , découle simplement de la définition de l'algorithme RM donnée plus haut, avec l'hypothèse que le paquet de BRM n est bien reçu par la source avant le début de l'intervalle $n+1$.

$R_e(n+1)$ se calcule en fonction de $R_e(n-1)$, $R_e(n)$, $d(n-1)$, $d(n)$, et $Op(n)$ étant le mode d'opération, égal à M1 ou M2, de la manière suivante :

Le nouveau taux d'émission $R_e(n+1)$ est alors donné par :

- Si $d(n) > T_q$, alors :

$$R_e(n+1) = R_e(n) - \theta \min(R_e(n-1), R_e(n)).$$

alors $Op(n+1) = M2$, sinon, $Op(n+1) = M1$.

- Si $d(n) \leq d(n-1)$ et $Op(n) = M1$, alors :

$$R_e(n+1) = R_e(n) + R_{inc}.$$

- Dans les autres cas, $R_e(n+1) = R_e(n)$.

Nous calculons enfin $Q(n+1)$ en fonction de $Q(n)$, $R_e(n)$, et $\mu(n)$. Lors du n ème intervalle, les données arrivent dans la file d'attente avec le taux $R_e(n)$, et le taux de service est égal à $\mu(n)$, pendant une durée égale à T . L'évolution de la longueur de la file d'attente est donc simplement donnée par :

$$Q(n+1) = Q(n) + T[R_e(n) - \mu(n)]$$

On voit que le processus défini par les grandeurs : $\{Q(n-1), Q(n), R_e(n-1), R_e(n), \mu(n-1), \mu(n), Op(n)\}$ est une chaîne de Markov sur un espace d'états inclus dans N^6 sur deux modes $\{M1; M2\}$ opérationnels. Une propriété remarquable de la chaîne de Markov ainsi définie est que les variables $R_e(n)$ et $Q(n)$ sont bornées [1]: il existe donc une valeur maximum finie du taux d'émission autorisé par l'algorithme, ainsi qu'une longueur maximum finie de la file d'attente. L'état initial de ce processus est tel que $Q(n) = 0$ et $R_e(n) = 0$ pour $t = 0$.

Dans notre modèle de l'algorithme RM, nous avons considéré que la détection de signes de congestion repose uniquement sur l'estimation $d_e(n)$ de délais de transmission, en fonction d'un seuil de tolérance T_q . L'objectif de l'algorithme est d'empêcher les réseaux d'atteindre un état de congestion. Si malgré tout la congestion se produit, un mécanisme de "time-out" est prévu pour réagir: si le paquet de BRM n'est pas reçu par la source au bout d'un certain temps T_{out} , le taux $R_e(n)$ est divisé par 2. Pour les conditions normales d'utilisation de l'algorithme, on suppose que le délai de retour ne dépasse jamais T_{out} , et le mécanisme de time-out n'intervient donc pas. Pour cette raison, nous ne considérons pas de mécanisme de time-out dans notre modèle. D'autre part, l'ajustement du taux $R_e(n)$ en fonction des informations du paquet de BRM se fait dès que ce paquet est reçu par la source. Nous avons supposé au contraire que le changement de $R_e(n)$ est effectué au début de l'intervalle de temps.

9- Simulations et exemples numériques

Nous présentons pour conclure quelques résultats de simulation qui illustrent le comportement de la chaîne définie précédemment. Cette simulation fait appel au langage Matlab dans sa version 6.1, sa souplesse et ses performances font de cet outil un logiciel d'avant-garde.

On commence donc par générer une séquence aléatoire représentant la marche du taux de service du serveur comprise entre deux valeurs : $\mu_m = 230 \times 10^3$ cellules/s, $\mu_M = 240 \times 10^3$ cellules/s.

Les paramètres du modèle sont les suivants: $T=150ms$, $Tq=20ms$, $R_{inc}=5$ cellules/s,. La durée de simulation est égale à 20s.

Nous considérons les valeurs suivantes pour : $\theta=0.5$, $\theta=0.8$ et $\theta=0.9$. Dans chaque cas, nous calculons la longueur $Q(n)$ de la file d'attente et le taux $R_e(n)$ comparé à la bande passante $\mu(n)$.

Les courbes de $Q(n)$ sur les figures 6, 7 et 8 mettent en évidence que le délai $d(n)$ oscille constamment entre 0 et Tq , avec des pics qui sont de l'ordre de grandeur de Tq . L'algorithme réagit donc manière satisfaisante l'augmentation de délai et empêche $d(n)$ de devenir trop grand, mais présente un problème d'oscillations, avec des variations rapides et répétées de $d(n)$.

Sur les figures 9, 10 et 11 qui représentent les courbes comparatives de $R_e(n)$ et $\mu(n)$, on observe que le mécanisme d'adaptation de taux fonctionne correctement : $R_e(n)$ suit $\mu(n)$ de relativement près, et cette adaptation est meilleure, dans ce modèle, pour un paramètre θ proche de 1.

Le calcul des meilleurs paramètres θ et R_{inc} pour un modèle donné est un problème difficile. D'une manière générale, il existe un compromis entre adaptabilité rapide face à des changements d'état du réseau et stabilité du système. Si l'on désire l'adaptabilité, rapide de l'algorithme, le taux $R_e(n)$ doit pouvoir varier rapidement pour que la source puisse s'adapter au plus vite aux conditions du réseau, et on doit donc avoir un taux de décroissance θ assez petit pour que $R_e(n)$ soit fortement réduit si nécessaire et une valeur R_{inc} assez grande pour que $R_e(n)$ augmente rapidement. Mais avec θ petit et R_{inc} grand, on peut avoir de brusques variations de $R_e(n)$ qui produisent des oscillations importantes de $Q(n)$. Inversement, si l'on désire un comportement stationnaire le plus régulier possible, par exemple dans le cas où $\mu(n)$ varie peu, le taux $R_e(n)$ lui aussi doit varier peu pour éviter les brusques oscillations, et il faut alors

choisir une valeur de θ proche de 1 et un paramètre R_{inc} petit. On peut aborder ce problème de compromis par optimisation des paramètres, ou par une adaptation dynamique.

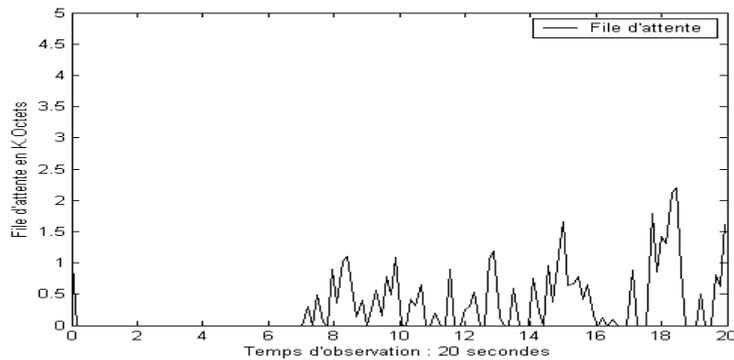


Figure 6 : File d'attente pour un rapport = 0.5.

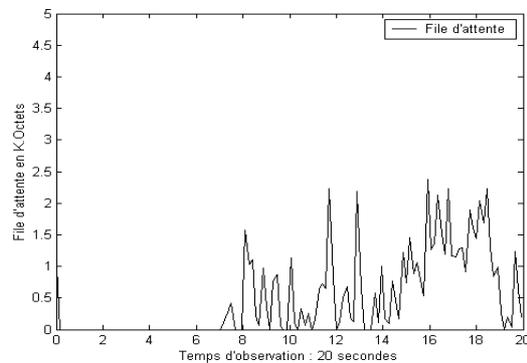


Figure 7 : File d'attente pour un rapport = 0.8.

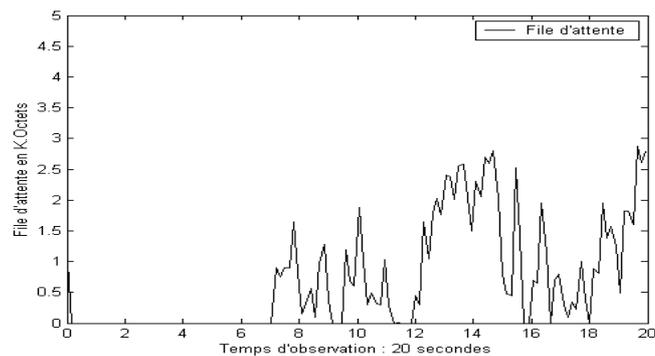


Figure 8 : File d'attente pour un rapport = 0.9.

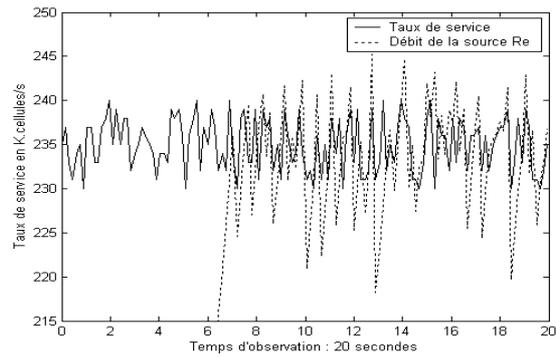


Figure 9 : Comparaison $\mu(n)$ et $R_e(n)$ pour un rapport = 0.5.

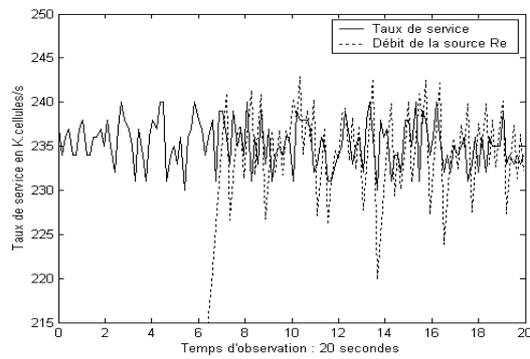


Figure 10 : Comparaison $\mu(n)$ et $R_e(n)$ pour un rapport = 0.8.

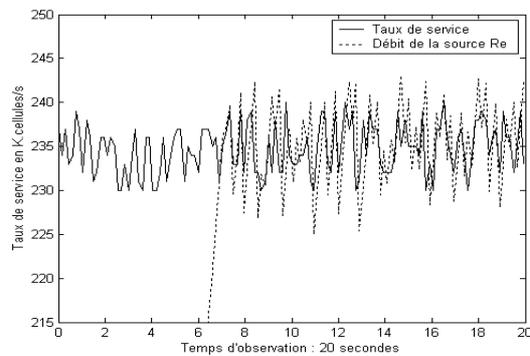


Figure 11 : Comparaison $\mu(n)$ et $R_c(n)$ pour un rapport = 0.9.

10- Conclusion

La congestion est un problème paralysant de l'ATM. On a proposé à travers cet article une approche algorithmique pour un contrôle réactif de la congestion. Cet algorithme se base sur des conclusions des articles de recherches [3,5,6]. L'approche des réseaux ATM, est réalisée par des hypothèses. L'originalité du travail, est cette modification ou adaptation, en partant du principe qu'un système de commutation par paquets mal conçu, crée des retards dans l'écoulement des données, très paralysant pour les communications interactives et temps réel, et qui provoque également des surcharges au niveau des nœuds d'accès.

L'évaluation des réseaux de transmission est en général, basée sur les processus de Markov et les files d'attente, et que seules des mesures statistiques dans des conditions optimales peuvent donner une idée sur l'ordre des grandeurs des paramètres à évaluer.

On peut ainsi poursuivre l'étude de l'algorithme par la recherche des paramètres optimaux pour un modèle donné. D'autres extensions envisageables sont les modèles avec plusieurs connexions multiplexées dans un même canal et contrôlées chacune par le RM. Il est donc important de comprendre l'intérêt de disposer d'une écriture explicite des valeurs des probabilités qui régissent ces phénomènes de blocage, tels que les retards, les surcharges, les congestions, etc.

L'étude des systèmes de commutation et des phénomènes de retards, ainsi que le contrôle de la congestion, est loin d'être simple, et fait l'axe d'orientation de plusieurs sujets de recherches.

Références :

- [1] G. Pujolle, *Les réseaux*, Editions Eyrolles (2001), Paris.
- [2] R. Händel, M.N. Huber, S. Schröder. *Comprendre ATM: concepts, protocols, applications*, Addison-Wesley (1995), France.
- [3] R. Jain, Congestion control and traffic management in ATM networks: recent advances and a survey. *Computer Networks and ISDN Systems*, 28 (13): 1723-1738, November 1996.
- [4] S. Floyd, K. Fall, Promoting the Use of end-to-end congestion control in the Internet, *IEEE/ACM Trans. on Networking*, pp . 458-472, Vol. 7, n° 4, August 1999.
- [5] E.P. Rathgeb, Modeling and Performance Comparaison of Policing Mechanisms for ATM Networks, *IEEE Journal on selected areas in communications*. pp. 325-334, Vol. 9, no. 3. April 1991.
- [6] K.P. Laberteaux, C.E. Rohrs (c.e.), P.J. Antsaklis, A Practical Controller for Explicit Rate Congestion Control. *IEEE Trans. on automatic Control*, pp . 960-978, Vol. 47, n°6, June 2002.
- [7] A. Kandel, O. Manor, Y. Klein, S. Fluss, ATM Traffic Management and congestion control using fuzzy logic. *IEEE Trans. Systems. man and cybernetics – Part C : Applications and reviews*. 29, n° 3, August 1999, pp. 474-480.
- [8] D. Ros, R. Marie, Simulation de modèles fluides pour réseaux haut débit. Rapport de Recherche, IRISA, n° 1224, Décembre 1998.
- [9] S. Fahmy, R. Jain, R. Goyal, B. Vandalore, Kalyanarman, Feed-back consolidation algorithms for ABR point-to-multipoint connections in ATM Networks, *Proc. Of IEEE INFOCOM*, pp. 1004-1013, April 1998.
- [10] A. Huang, C. Moreland, I; Wright, Advanced Traffic Management for Multiservice ATM Networks. *N.E.T. White Paper*, Network Equipment Technology, Inc. USA 1998.

[11] S. Fahmy, R. Jain, R. Goyal, B. Vandalore, Proposed modifications to the baseline text end living list on multipoint ABR behaviour, ATM Forum contribution 98-0150, February 1998.

[12] X. Zhang, K. Shin, D. Saha, D. Kandlur, Scalable Flow Control for Multicast ABR Services in ATM Networks. *IEEE/ACM Trans. on Networks*, Vol. 9, n°1, February 2002

[13] P. Chandra, A. Fisher, C. Kosak and P. Steenkiste, "Experimental Evaluation of ATM Congestion Control Mechanisms." In proceedings - INFOCOM IEEE 1997.